

# JAVA SCRIPT





# JavaScript Output

## JavaScript Display Possibilities

JavaScript can "display" data in different ways:

- Writing into an HTML element, using **innerHTML**.
- Writing into the HTML output using **document.write()**.
- Writing into an alert box, using **window.alert()**.
- Writing into the browser console, using **console.log()**.

## Using innerHTML

- To access an HTML element, JavaScript can use the **document.getElementById(id)** method.
- The **id** attribute defines the HTML element. The **innerHTML** property defines the HTML content:

# Using innerHTML

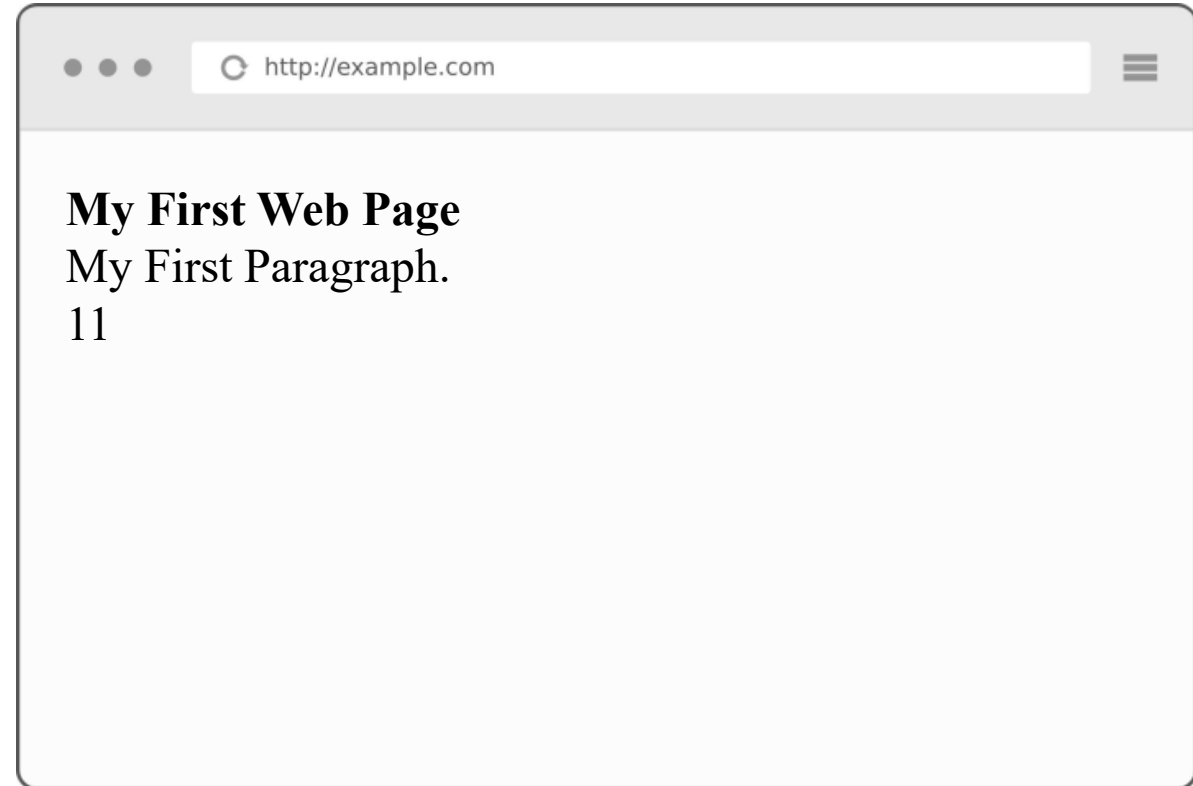
```
<!DOCTYPE html>
<html>
<body>

<h2>My First Web Page</h2>
<p>My First Paragraph.</p>

<p id="demo"></p>

<script>
document.getElementById("demo").innerHTML = 5 + 6;
</script>

</body>
</html>
```



Changing the innerHTML property of an HTML element is a common way to display data in HTML.



# Using document.write()

```
<!DOCTYPE html>
<html>
<body>

<h2>My First Web Page</h2>
<p>My first paragraph.</p>

<p>Never call document.write after the document has finished
loading.
It will overwrite the whole document.</p>

<script>
document.write(5 + 6);
</script>

</body>
</html>
```

# Using document.write()

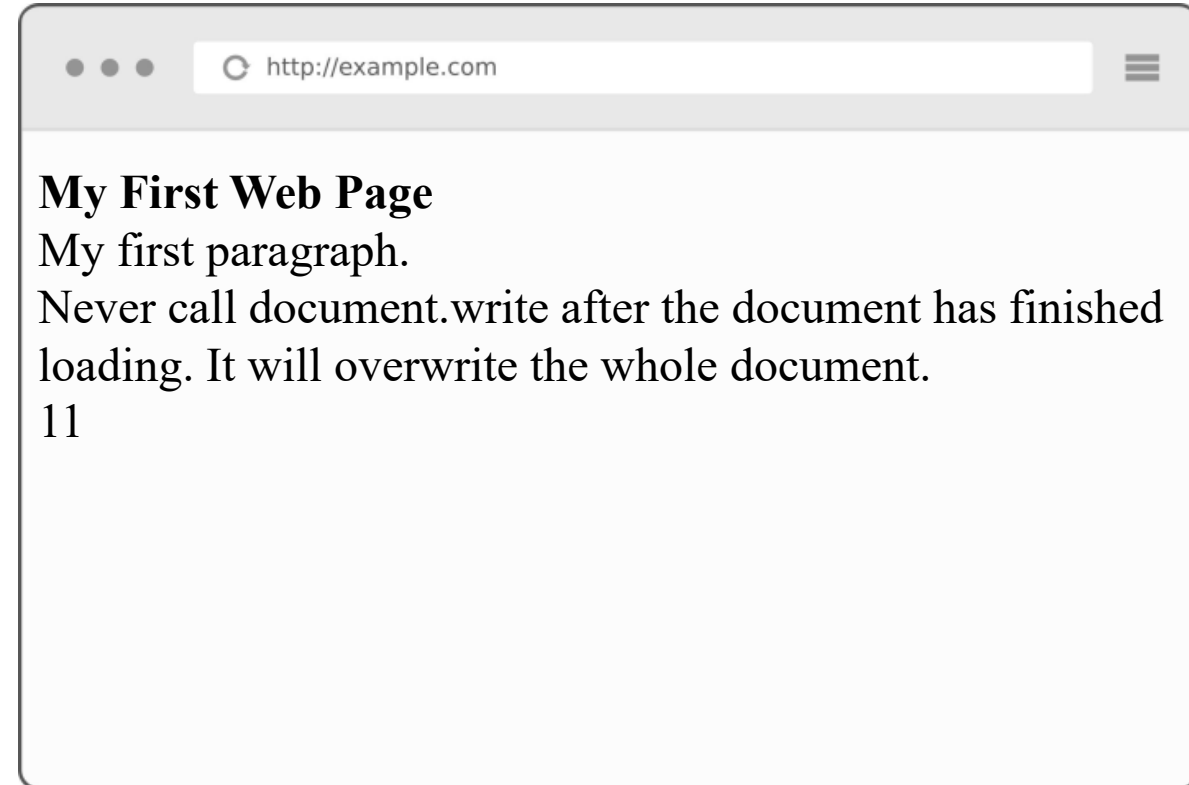
```
<!DOCTYPE html>
<html>
<body>

<h2>My First Web Page</h2>
<p>My first paragraph.</p>

<p>Never call document.write after the document has finished
loading.
It will overwrite the whole document.</p>

<script>
document.write(5 + 6);
</script>

</body>
</html>
```



Using document.write() after an HTML document is loaded, will **delete all existing HTML**:

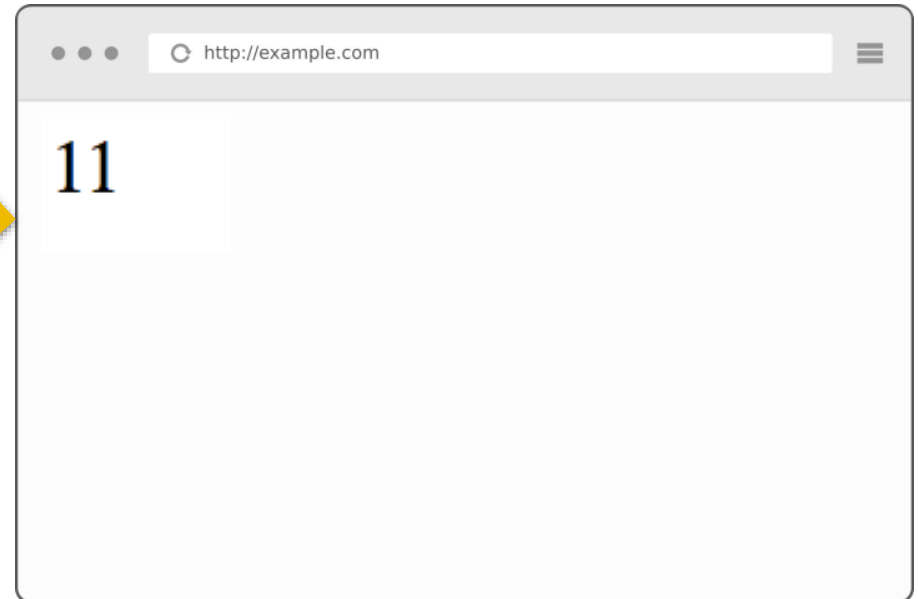
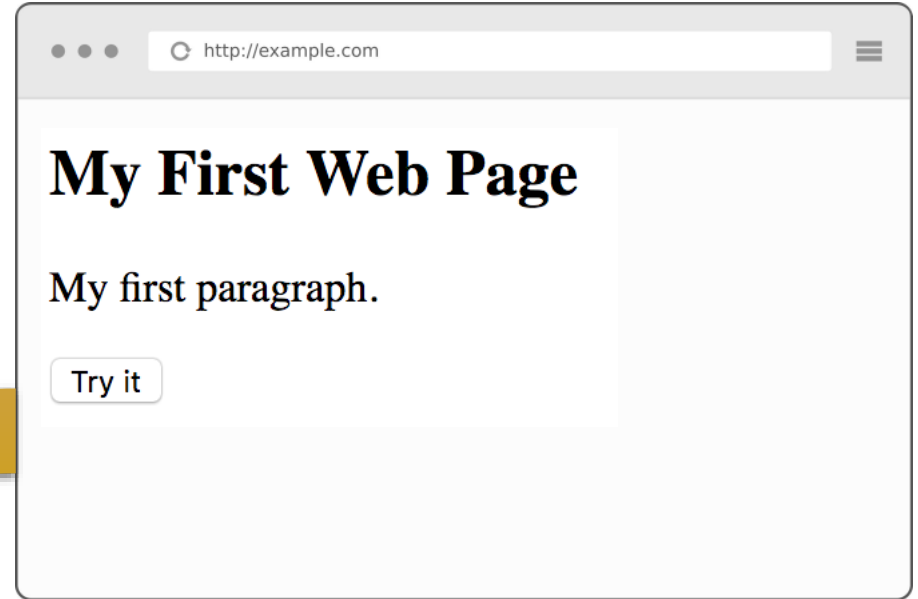
## Using Document.write

```
<!DOCTYPE html>
<html>
<body>

<h2>My First Web Page</h2>
<p>My first paragraph.</p>

<button type="button"
onclick="document.write(5 + 6)">
Try it
</button>

</body>
</html>
```





## Using Document.write

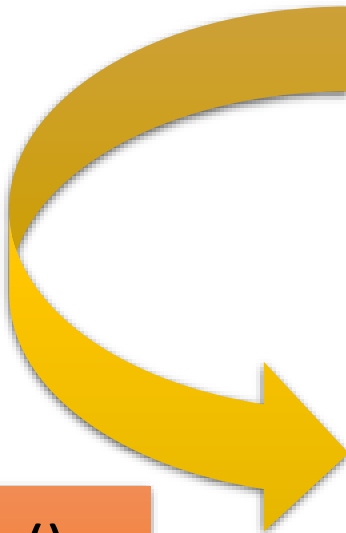
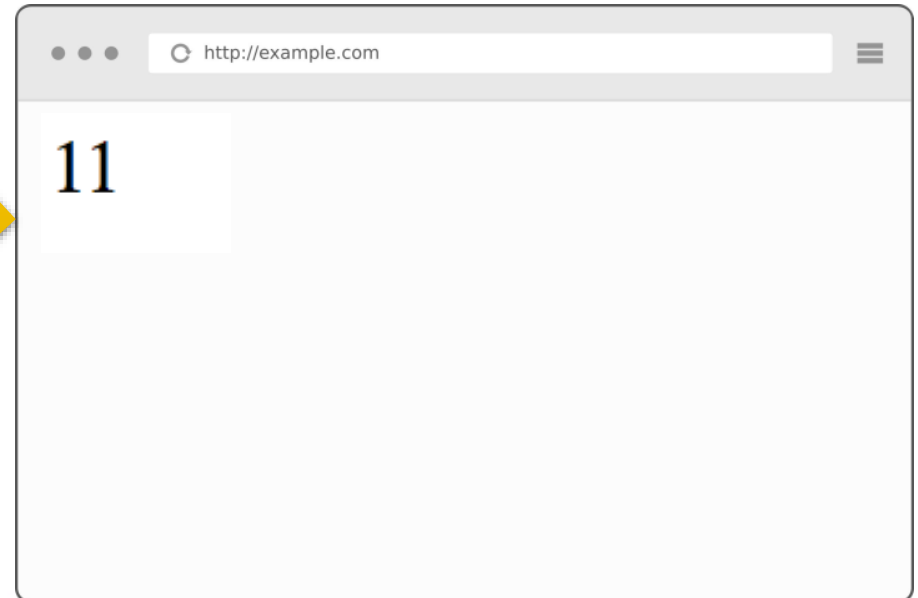
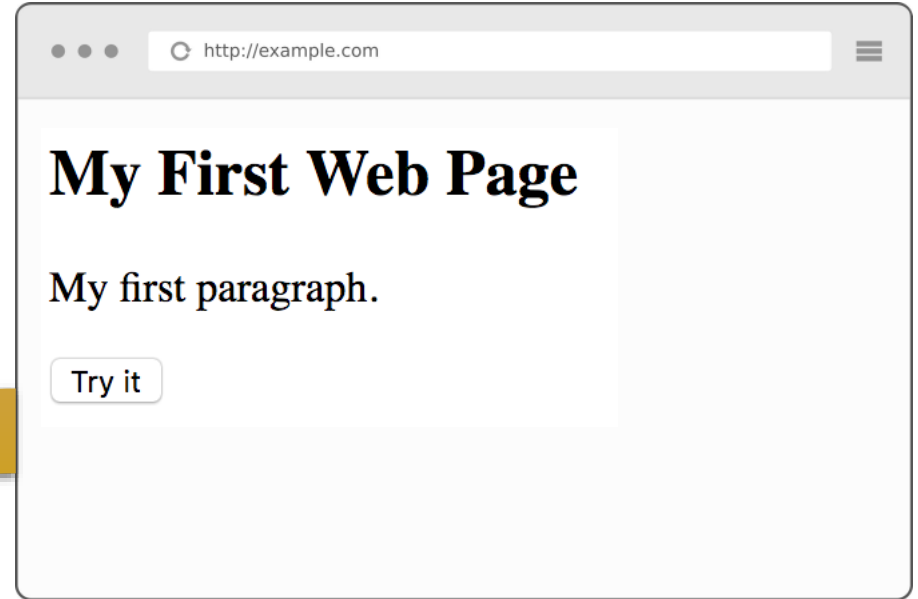
```
<!DOCTYPE html>
<html>
<body>

<h2>My First Web Page</h2>
<p>My first paragraph.</p>

<button type="button"
onclick="document.write(5 + 6)">
Try it
</button>

</body>
</html>
```

The document.write() method should only be used for testing.





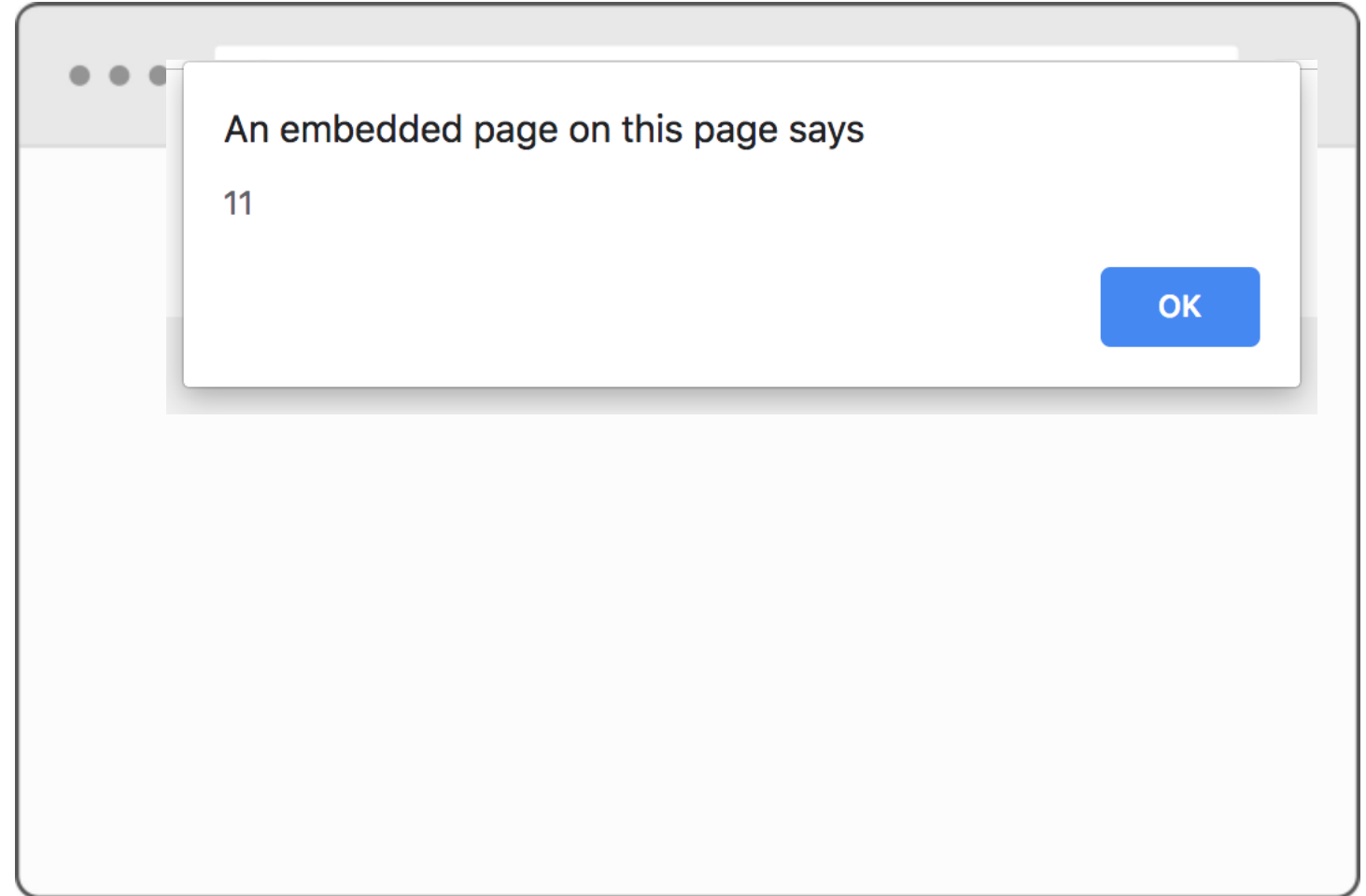
# Using window.alert()

```
<!DOCTYPE html>
<html>
<body>

<h2>My First Web Page</h2>
<p>My first paragraph.</p>

<script>
window.alert(5 + 6);
</script>

</body>
</html>
```



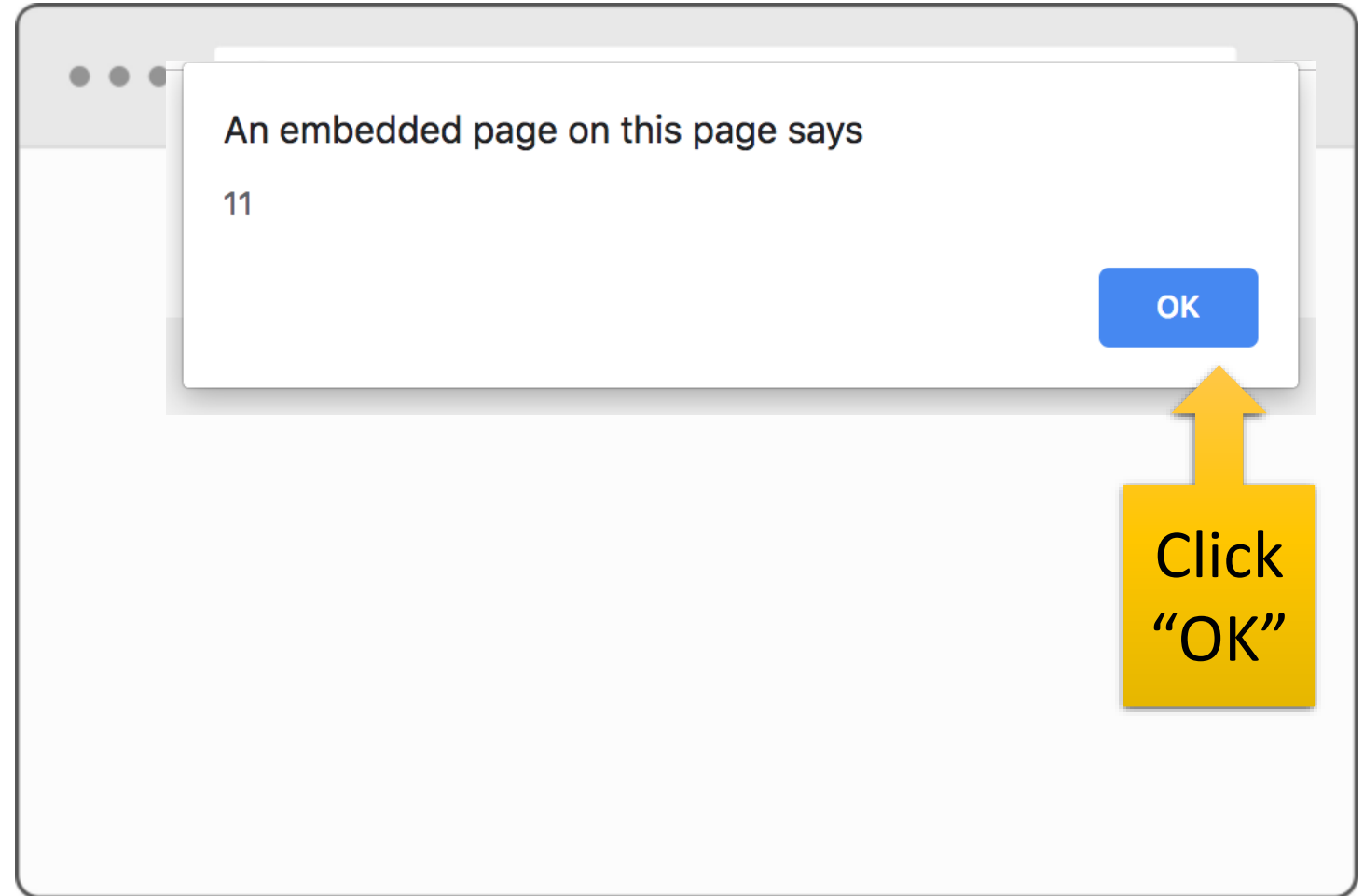
# Using window.alert()

```
<!DOCTYPE html>
<html>
<body>

<h2>My First Web Page</h2>
<p>My first paragraph.</p>

<script>
window.alert(5 + 6);
</script>

</body>
</html>
```





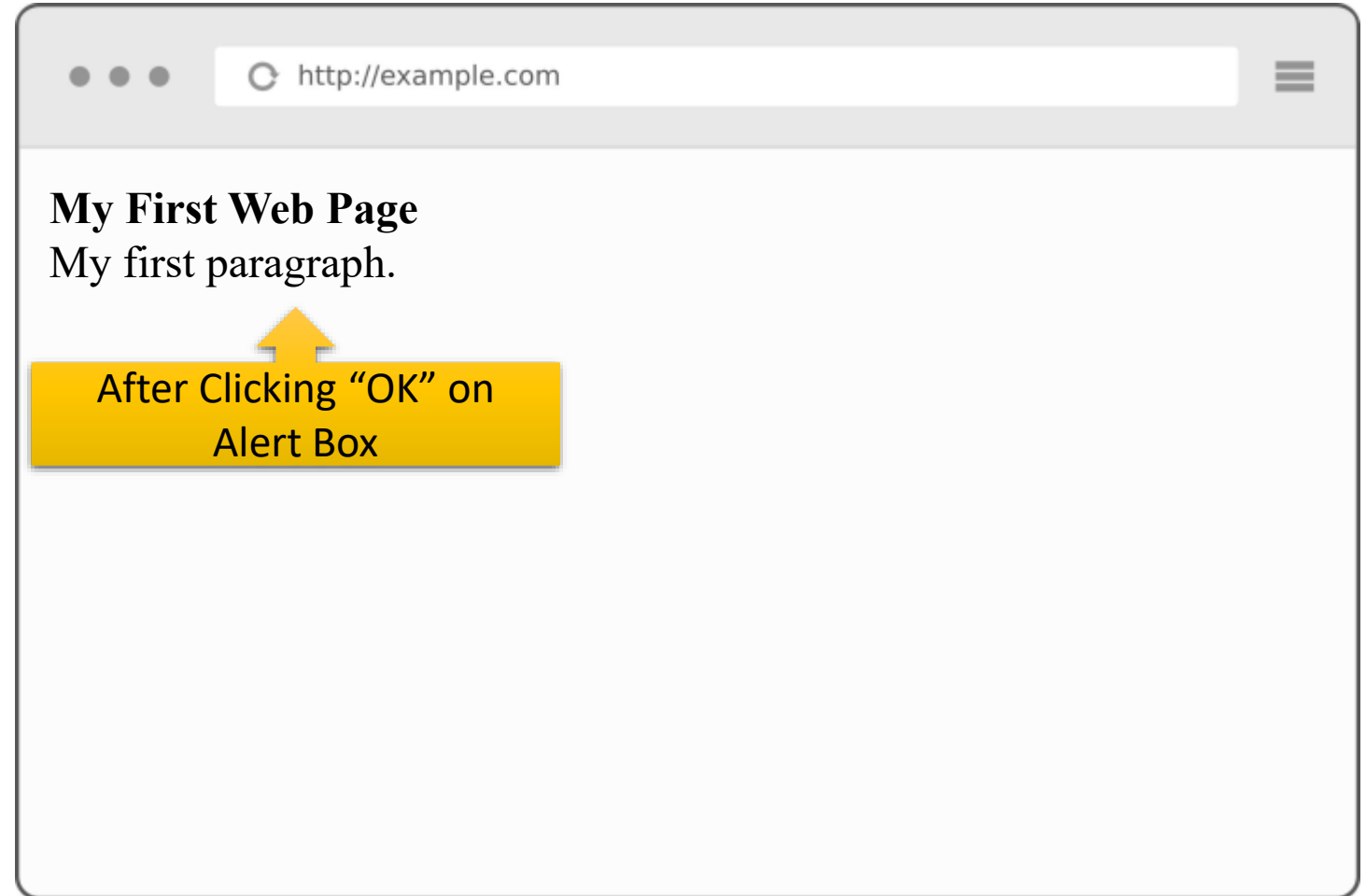
# Using window.alert()

```
<!DOCTYPE html>
<html>
<body>

<h2>My First Web Page</h2>
<p>My first paragraph.</p>

<script>
window.alert(5 + 6);
</script>

</body>
</html>
```



# Using console.log()

For debugging purposes, you can use the **console.log()** method to display data

```
<!DOCTYPE html>
<html>
<body>

<h2>Activate debugging with F12</h2>

<p>Select "Console" in the debugger menu. Then click Run again.</p>

<script>
console.log(5 + 6);
</script>

</body>
</html>
```



```
<!DOCTYPE html>
<html>
<body>

<h2>Activate debugging with F12</h2>

<p>Select "Console" in the debugger
menu. Then click Run again.</p>

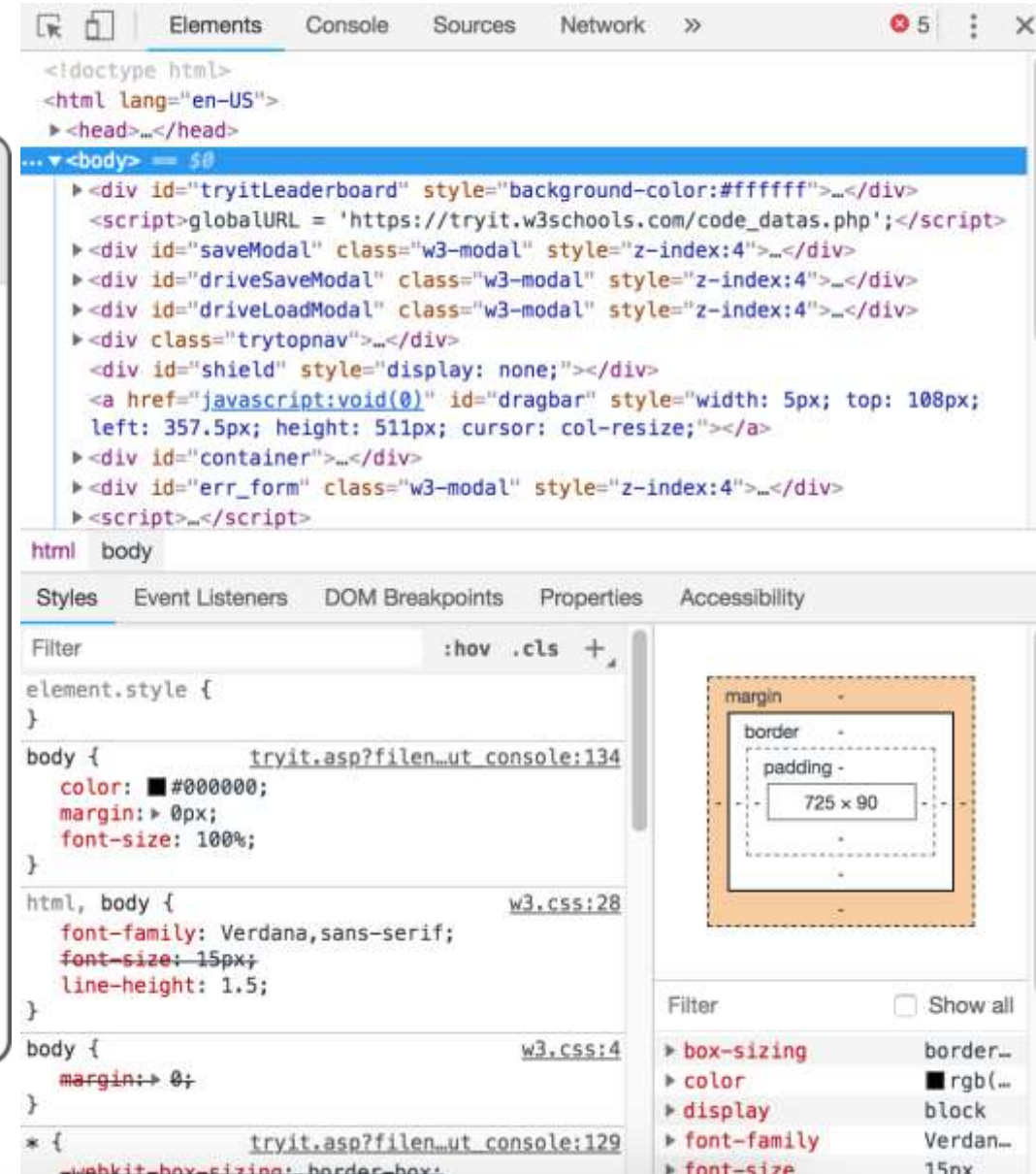
<script>
console.log(5 + 6);
</script>

</body>
</html>
```



## Activate debugging with F12

Select "Console" in the debugger menu. Then click Run again.



The screenshot shows a web browser window with the address bar displaying `http://example.com`. The developer console is open, showing the following HTML code:

```
<!doctype html>
<html lang="en-US">
  <head>...</head>
  <body>
    <div id="tryitLeaderboard" style="background-color:#ffffff">...</div>
    <script>globalURL = 'https://tryit.w3schools.com/code_datas.php';</script>
    <div id="saveModal" class="w3-modal" style="z-index:4">...</div>
    <div id="driveSaveModal" class="w3-modal" style="z-index:4">...</div>
    <div id="driveLoadModal" class="w3-modal" style="z-index:4">...</div>
    <div class="trytopnav">...</div>
    <div id="shield" style="display: none;"></div>
    <a href="javascript:void(0)" id="dragbar" style="width: 5px; top: 108px; left: 357.5px; height: 511px; cursor: col-resize;"></a>
    <div id="container">...</div>
    <div id="err_form" class="w3-modal" style="z-index:4">...</div>
    <script>...</script>
```

The console also shows the following CSS styles:

```
body {
  color: #000000;
  margin: 0px;
  font-size: 100%;
}

html, body {
  font-family: Verdana,sans-serif;
  font-size: 15px;
  line-height: 1.5;
}

body {
  margin: 0;
}

* {
  -webkit-box-sizing: border-box;
```

On the right side of the console, there is a visual representation of the box model for the selected element, showing the margin, border, padding, and the content area (725 x 90).



## Activate debugging with F12

Select "Console" in the debugger menu.  
Then click Run again.

The screenshot shows the Chrome DevTools Console with the following error messages:

- Failed to load resource: `static.h-bid.com/w3s...-w3schools.min.js:1`  
`net::ERR_BLOCKED_BY_CLIENT`
- Failed to load resource: `cmp.stub.js:1`  
`net::ERR_BLOCKED_BY_CLIENT`
- Failed to load resource: `integrator.js:1`  
`net::ERR_BLOCKED_BY_CLIENT`
- Failed to load resource: `integrator.js:1`  
`net::ERR_BLOCKED_BY_CLIENT`
- Uncaught TypeError: `snhb.startAuction is not a function`  
at `HTMLButtonElement.onclick` (`tryit.asp?filename=t... output console:433`)

